REMARKS

In view of the above amendments and the following remarks, reconsideration of the objection and rejections and further examination are respectfully requested.

The specification and abstract have been reviewed and revised to improve their U.S. form. The amendments to the specification and abstract have been incorporated into a substitute specification and abstract. Attached are two versions of the substitute specification, a marked-up version showing the revisions, as well as a clean version. No new matter has been added.

Claims 1-9 were objected to as containing informalities. Specifically, claims 1-9 were objected to for failing to establish proper antecedent basis for each limitation recited therein. In view of the Examiner's objections, claims 1-9 have been amended to establish proper antecedent basis for each limitation recited therein. Thus, it is respectfully submitted that this objection is no longer applicable.

Further, claims 1-10 have also been amended to make a number of editorial revisions thereto. These revisions have been made to slightly clarify the scope of the claims and place the claims in better U.S. form.

In addition, claim 11 has been added to depend from independent claim 1. Support for new claim 11 can be found in, at least, paragraph [0023] of the specification of the present application.

Claims 1, 4-7 and 9 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Parry et al. (U.S. 6,748,481 B1) in view of Paul Sheer's "Rute User's Tutorial and Exposition." Further, claims 2-3, 8 and 10 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Parry and Paul Sheer's "Rute User's Tutorial and Exposition" and further in view of Barton et al. (U.S. 6,233,389 B1). The above-mentioned rejections are clearly inapplicable to amended claims 1-10 and new claim 11 for the following reasons.

Please note that a brief description of the arrangement and advantages of the present invention as recited in amended independent claim 1 will now be provided with reference to various portions of the present application. However, reference to any specific portions of the present application is provided only for illustrative purposes, and is not intended to otherwise limit the scope of the claims to any particular embodiment.

As illustrated in Figure 1, amended independent Claim 1 is now directed a stream data processing apparatus that comprises, in part, a transmitting-end processing section (13), a

receiving-end processing section (14), a control section (11), a data temporary storage section (22), an empty data storage section (23), and a connection management section (21). The above-recited sections are interconnected and configured such that:

(A) if a change signal (31, 32) is transmitted from the control section (11) (i) to the transmitting-end processing section (13), the transmitting-end processing section (13) outputs a transmitting-end clear request (41) to the connection management section (21), and (ii) to the receiving-end processing section (14), the receiving-end processing section (14) outputs a receiving-end clear request (42) to the connection management section (21); and

(B) the connection management section (21) is operable to switch both (i) a write destination for the processed data transmitted from the transmitting-end processing section (13), and (ii) a read source of any data received by the receiving-end processing section (14), between the data temporary storage section (22) and the empty data storage section (23), depending on whether the connection management section (21) is in (i) a normal operation state, (ii) a receiving-end clear wait state which exists after the transmitting-end clear request is received by the connection management section and until the receiving-end clear request is received by the connection management section, or (iii) a transmitting-end clear wait state which exists after the receiving-end clear request is received by the connection management section and until the transmitting-end clear request is received by the connection management section.

In other words, claim 1 recites two processing sections (i.e., transmitting-end and receiving-end) which transmit/receive data to/from, a data temporary storage section and/or an empty data storage section in various combinations based on a change signal, clear request signals, and a state of a connection management section. Specifically, the transmitting-end processing section outputs a transmitting-end clear request to the connection management section if a change signal is received from a control section and the receiving-end processing section outputs a receiving-end clear request to the connection management section if a change signal is received from the control section. Finally, based on the state of the connection management section, which is determined according to the receipt (and order of receipt) of both the clear requests, the connection management section switches a write destination and a read

source of any data transmitted/received to/from the two processing sections to the data temporary storage section and the empty data storage section.

Please note that a benefit of the above-described configuration is that each processing section outputs a clear request which is then used by the connection management section to switch the write destination and the read source of the two processing sections, wherein the processing of the two processing sections is continuous and uninterrupted since the processing sections continue to process the data and the connection management section controls the writing destination and the reading source of the data. The Parry reference fails to disclose or suggest the above-mentioned features and advantages of independent claim 1.

In contrast, the Parry reference teaches a streaming information appliance. Specifically, the streaming information appliance of Parry includes a buffer, a writer module (i.e., processing section) which receives blocks of streaming information and writes the blocks to the buffer and a read module (i.e., processing section) which selectively reads the blocks from the buffer (see abstract). In addition, Parry teaches that the reader module can pause or fast-forward the data stored in the buffer, wherein the writer module can be blocked from fast-forwarding if the writer module catches up to the reader module, or, in the alternative the reader module can be blocked until appropriate data is available (see col. 13, lines 15-33).

This "blocking" disclosed in Parry, as discussed above, is controlled, in part, directly by both the writing module and the reading module. The writing function of the writing module is illustrated in Figs. 9A and 9B, wherein the various steps of the writing include locking, advancing a tail pointer, blocking, unblocking, and waiting based on a multitude of conditions, wherein based on those conditions reading and/or writing may be blocked by the writing module itself (see steps 301-310 of Fig. 9A and steps 311-317 of Fig. 9B). Likewise, the reading function of the reading module is similarly illustrated in Figs. 11A and 11B, wherein based on a multitude of conditions reading and/or writing may be blocked by the reading module itself.

In addition, Parry teaches that a buffer IO layer 200 implements a synchronization algorithm to instruct the reader and writer modules to implement the "blocking" described above, wherein the present status/operation of the reader and writer modules is critical in determining the "blocking" according to the algorithm (see col. 8, lines 54-57 and 64-67).

In view of the above disclosures of the Parry reference, the following differences between the present invention as recited in independent claim 1 and the Parry reference become evident.

Parry teaches that a synchronization algorithm instructs the writing/reading modules to lock, advance a tail pointer, block, unblock, and wait based on a multitude of conditions for the purpose of controlling the writing/reading of the data (i.e., the writing/reading modules are required to process information based on multiple conditions in order to control the reading/writing of the data). However, Parry does not disclose or suggest changing the write destination and the read source of data of the two processing sections simply based on the transmission of the change signal to the two processing sections from the control section, wherein each processing section outputs a clear request which is then used to switch (i.e., configure) the read/write destinations, by the connection management section, of the two processing sections.

In other words, having the two processing sections simply output a clear request based on a change signal, wherein the combination of clear requests is used to determine the write/read destination/source of the two processing sections (i.e., the configuration recited in claim 1), is not the same as or even similar to write/read modules (i.e., processing sections) which, based on a multitude of conditions, are required to process and output various signals to implement the blocking, wherein the status of each module is required to be known by the other modules in order to implement the blocking (i.e., as disclosed in the Parry reference). Simply stated, the connection management section of claim 1 switches the write destination and read source, a feature which is not disclosed or suggested by the writing/reading/blocking being determined and controlled, in part, by the write/read modules themselves, since, for example, the location which the data is transmitted/received to/from is controlled by the connection management section and not controlled by the processing section(s) itself/themselves.

In view of the above, it is respectfully submitted that the Parry reference does not anticipate the invention as recited in amended claim 1. Furthermore, Parry does not suggest the above-discussed limitations of claim 1. Therefore, it would not have been obvious to one of ordinary skill in the art to modify the Parry reference so as to obtain the invention of amended claims or the claims that depend therefrom. Accordingly, it is respectfully submitted that amended claims 1-11 are clearly allowable over Parry for the reasons discussed above.

Regarding Paul Sheer's "Rute User's Tutorial and Exposition" and Barton, these references do not disclose or suggest any of the above-discussed features of independent claim 1 which are lacking from the Parry reference. As a result, the above-mentioned references,

individually or collectively, fail to anticipate or render obvious amended independent claim 1 or the claims that depend therefrom.

In view of the above amendments and remarks, it is submitted that the present application is now in condition for allowance and an early notification thereof is earnestly requested. The Examiner is invited to contact the undersigned by telephone to resolve any remaining issues.

Respectfully submitted,

Satoshi INAMI et al.

By: _____
W. Douglas Hahm
Registration No. 44,142
Attorney for Applicants

WDH/ALD/nrj
Washington, D.C. 20006-1021
Telephone (202) 721-8200
Facsimile (202) 721-8250
June 6, 2007

Version with Markings to Show Changes Made

~~TITLE OF THE INVENTION~~

STREAM DATA PROCESSING APPARATUS

BACKGROUND OF THE INVENTION

1. Field of the Invention

[0001] The present invention relates to a stream data processing apparatus which performs multiple steps of processing for stream data. More particularly, the present invention relates to a stream data processing apparatus in which data is transmitted or received between a plurality of processing sections while achieving synchronization therebetween.

2. Description of the ~~Background~~ Related Art

[0002] Stream data processing apparatuses for performing multiple steps of processing ~~for~~ stream data such as audio data or video data are known. A typical stream data processing apparatus comprises an input section, a processing section, and an output section, such that the processing section performs a predetermined process for inputted stream data, and outputs the processing results thus obtained. In many cases, a stream data processing apparatus performs a processing consisting of a plurality of steps for the inputted stream data. ~~To this end~~Accordingly, the stream data processing apparatus comprises a plurality of processing sections for executing the processes of the respective steps, and data is transmitted or received between such processing sections.

Thus, the stream data processing apparatus executes multiple steps of processing in a pipeline manner for the inputted stream data. Sometimes instructions such as interruption of a process or a seek (i.e., commencing processing at a different position in a given stream of data which is being processed) may be given to the stream data processing apparatus. In order to ensure proper operation when receiving such instructions, the stream data processing apparatus performs exchange of data while achieving synchronization between the plurality of processing sections.

[0003] Japanese Patent No. 3095290 discloses a method, performed ~~in~~ by a stream data processing apparatus, of exchanging data while achieving synchronization between a plurality of processing sections. According to this method, each processing section acquires via communications the state of another processing section with which it is exchanging data, and switches its own operation in accordance with the state of the other processing section. For example, when a seek instruction is given to the stream data processing apparatus, the processing section which is at the receiving end of data must postpone the receiving of data until the processing section at the transmitting end of data begins to transmit a processing result for the data corresponding to the new position. In order to realize such a function, each processing section controls its own operation so as not to read any improper data, in accordance with the content of the received data, the progress of the process, and the state of the other

processing section. Thus, by properly exchanging data between a plurality of processing sections while achieving synchronization therebetween, the stream data processing apparatus can operate with a good overall integrity.

[0004] However, achieving synchronization between a plurality of processing sections using the above-described method has the following problems. Since each processing section needs to control its own operation in accordance with the data content, progress of the processing, and the state of another processing section, the structure of each processing section becomes complicated. When implementing each processing section in software, in particular, the structure of the software component constituting each processing section becomes complicated. Moreover, since a unified interface is not used between the respective processing sections in the above-described method, it is difficult to port the software component constituting each processing section into another system.

SUMMARY OF THE INVENTION

[0005] Therefore, an object of the present invention is to provide a stream data processing apparatus in which data is exchanged between a plurality of processing sections while achieving synchronization therebetween, such that each processing section is capable of performing the exchange of data without having to pay attention to the data content, progress of the processing,

and the state of another processing section. Another object of the present invention is to unify the interface between the processing sections included in the stream data processing apparatus, thereby simplifying the structure of each processing section and improving the reusability of each processing section.

[0006] The stream data processing apparatus according to the present invention comprises a transmitting-end processing section, a receiving-end processing section, a control section, a data temporary storage section, an empty data storage section, and a connection management section, and performs multiple steps of processing for stream data. The transmitting-end processing section performs a process of one of the multiple steps of processing. The receiving-end processing section performs a process of a next one of the multiple steps of processing. The connection management section employs the data temporary storage section as a buffer for allowing the data transmitted from the transmitting-end processing section to be received by the receiving-end processing section. If it becomes necessary to change the subject of processing due to a seek or a change of the stream data, each processing section outputs a clear request to the connection management section in response to an instruction from the control section. The connection management section switches a write destination and a read source depending on whether the connection management section is in a normal operation state, a receiving-end clear wait state which exists after the transmitting-end clear

request is received and until the receiving-end clear request is received, or a transmitting-end clear wait state which exists after the receiving-end clear request is received and until the transmitting-end clear request is received. As a destination for such switching, an empty data storage section, which erases any data written thereto in response to a data write and returns empty data in response to a data read, is used.

[0007] As a first example method of switching, the connection management section may select the empty data storage section as the read source in the receiving-end clear wait state, and select the empty data storage section as the write destination in the transmitting-end clear wait state. When this method of switching is adopted, the connection management section erases the data stored in the data temporary storage section if ~~either~~ a clear request is received in the normal operation state.

[0008] As a second example method of switching, the connection management section may select the empty data storage section as the write destination in the transmitting-end clear wait state, and in the receiving-end clear wait state, regard as old data any data that is stored in the data temporary storage section when the transmitting-end clear request has been received, select as the write destination a region in the data temporary storage section where the old data is not stored, and select as the read source a region in the data temporary storage section where the old data is stored while the old data is present (and select the empty data

storage section as the read source once the old data is no longer present). When this method of switching is adopted, the connection management section erases the data stored in the data temporary storage section if the receiving-end clear request is received in the normal operation state, and erase the old data if the receiving-end clear request is received in the receiving-end clear wait state.

[0009] The transmitting-end processing section and the receiving-end processing section may output the transmitting-end clear request and the receiving-end clear request and perform transmission and reception of data by using a data transmission/reception section which provides an accessing function to the connection management section.

[0010] The connection management section may be structured to be capable of selecting, if the data transmitted from the transmitting-end processing section cannot be written to the data temporary storage section, whether to perform a process of immediately notifying an error to the transmitting-end processing section, or to perform a process of waiting until it becomes possible to write data to the data temporary storage section and notifying to the transmitting-end processing section a result of writing data to the data temporary storage section. Alternatively, the connection management section may be structured to be capable of selecting, if data to be received by the receiving-end processing section cannot be read from the data temporary storage section,

a process of immediately notifying an error to the receiving-end processing section, or a process of waiting until it becomes possible to read data from the data temporary storage section and notifying to the receiving-end processing section a result of reading data from the data temporary storage section.

[0011] The stream data processing apparatus may further comprise a data input section via which to input the stream data, and/or a data output section for outputting a result of performing the multiple steps of processing ~~for~~ the stream data. The data input section may input the stream data from a removable recording medium. The data output section may output the result of performing the multiple steps of processing to a removable recording medium.

[0012] These and other objects, features, aspects and advantages of the present invention will become more apparent from the following detailed description of the present invention when taken in conjunction with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0013] FIG. 1 is a block diagram illustrating the structure of a stream data processing apparatus according to first and second embodiments of the present invention;

FIG. 2 is a flowchart illustrating the operation of the stream data processing apparatus according to the first and second embodiments of the present invention in the case where a commencement of processing is instructed in a processing stopped

state;

FIG. 3 is a flowchart illustrating the operation of the stream data processing apparatus according to the first and second embodiments of the present invention in the case where a change of the subject of processing is instructed in a normal operation state;

FIG. 4 is a flowchart illustrating the operation of a connection management section of the stream data processing apparatus according to the first embodiment of the present invention;

FIG. 5 is a diagram illustrating a manner in which data is transmitted or received in the stream data processing apparatus according to the first and second embodiments of the present invention in a normal operation state;

FIG. 6 is a diagram illustrating a manner in which data is transmitted or received in the stream data processing apparatus according to the first embodiment of the present invention in a receiving-end clear wait state;

FIG. 7 is a diagram illustrating a manner in which data is transmitted or received in the stream data processing apparatus according to the first and second embodiments of the present invention in a transmitting-end clear wait state;

FIG. 8 is a table illustrating the operation of the connection management section of the stream data processing apparatus according to the first embodiment of the present

invention;

FIG. 9 is a flowchart illustrating the operation of a connection management section of the stream data processing apparatus according to the second embodiment of the present invention;

FIG. 10 is a diagram illustrating a manner in which data is transmitted or received in the stream data processing apparatus according to the second embodiment of the present invention in a receiving-end clear wait state, in the case where old data is present;

FIG. 11 is a diagram illustrating a manner in which data is transmitted or received in the stream data processing apparatus according to the second embodiment of the present invention in a receiving-end clear wait state, in the case where old data is no longer present;

FIG. 12 is a table illustrating the operation of the connection management section of the stream data processing apparatus according to the second embodiment of the present invention; and

FIG. 13 is a block diagram illustrating the structure of a stream data processing apparatus according to a variant of the present invention.

DETAILED DESCRIPTION OF THE ~~PREFERRED EMBODIMENTS~~INVENTION

(first embodiment)

[0014] FIG. 1 is a block diagram illustrating the structure of a stream data processing apparatus according to the first embodiment of the present invention. The stream data processing apparatus 10 shown in FIG. 1 comprises a control section 11, a data input section 12, a first processing section 13, a second processing section 14, a data output section 15, a connection management section 21, a data temporary storage section 22, an empty data storage section 23, a data transmission section 24, and a data reception section 25. The stream data processing apparatus 10 performs two steps of processing ~~for~~ inputted stream data. As used herein, "stream data" refers to data which is inputted/outputted or processed in a chronological order, e.g., audio data or video data.

[0015] The control section 11 is a means for controlling the operation of the stream data processing apparatus 10. When a command is inputted from a user, or when an abnormality occurs in the apparatus, for example, the control section 11 outputs control signals 31 and 32, respectively, to the first processing section 13 and the second processing section 14.

[0016] The data input section 12 is a means by which to input stream data. The data input section 12 may be a storage device such as a hard disk drive or a memory, for example. The data input section 12, when composed of such a storage device, stores stream data in a predetermined format, and outputs the data contained in the stream data in a chronological order. Alternatively, the

data input section 12 may be a communication controlling device which performs communications with, e.g., a server (not shown) which is coupled to data input section 12 via a network. The data input section 12, when composed of such a communication controlling device, receives the data contained in stream data in a chronological order via communications, and outputs the received data in the order in which they are received.

[0017] The first and second processing sections 13 and 14 perform two steps of processing ~~for~~ the inputted stream data. More specifically, the first processing section 13 performs a first step of processing ~~for~~ the data which has been inputted in a chronological order via the data input section 12, and transmits the processed data to the second processing section 14. The second processing section 14 receives the data which has been transmitted from the first processing section 13, and performs a second step of processing ~~for~~ the received data. In the structure shown in FIG. 1, the first processing section 13 functions as a transmitting-end processing section, and the second processing section 14 functions as a receiving-end processing section.

[0018] The content of the processes performed by the first processing section 13 and the second processing section 14 may be arbitrary. Examples of the first processing section 13 and the second processing section 14 may be MPEG (Moving Pictures Experts Group) encoders, MPEG decoders, D/A converters for converting digital audio data to analog audio data, source filters

used for obtaining data from a memory or the like, or renderer filters for adjusting the output timing of video signals.

[0019] The data output section 15 is an output means for outputting the processing result obtained by subjecting the inputted stream data to the two-step processing. The data output section 15 may be an output device such as a display device or a loudspeaker. Alternatively, the data output section 15 may be a storage device such as a hard disk drive or a memory. The data output section 15, when composed of such a storage device, stores the chronologically-outputted processing results in a predetermined format. Alternatively, the data output section 15 may be a communication controlling device which performs communications with, e.g., a server (not shown) which is coupled to the data output section 15 via a network. The data output section 15, when composed of such a communication controlling device, transmits via communications the aforementioned processing results in the order in which they are outputted.

[0020] The connection management section 21 utilizes the data temporary storage section 22 to allow the data which has been transmitted from the first processing section 13 to be received by the second processing section 14. More specifically, in response to a transmission request from the first processing section 13, the connection management section 21 writes to the data temporary storage section 22 the data transmitted from the first processing section 13. In response to a reception request

from the second processing section 14, the connection management section 21 allows the second processing section 14 to receive the data which has been read from the data temporary storage section 22. The data temporary storage section 22 temporarily stores the data which has been transmitted from the first processing section 13. The data temporary storage section 22 may be composed of, for example, a storage device such as a hard disk drive or a memory. In order to ensure efficient management and storage of data, a buffer structure may be adopted for the data temporary storage section 22, e.g., queue, double buffer, or a ring buffer.

[0021] If it is impossible to write to the data temporary storage section 22 the data which has been transmitted from the first processing section 13 (e.g., if the data temporary storage section 22 is full), the connection management section 21 waits until it becomes possible to write data to the data temporary storage section 22, and notifies to the first processing section 13 a result of writing data to the temporary storage section 22. If it is impossible to read from the data temporary storage section 22 any data which is to be received by the second processing section 14 (e.g., if the data temporary storage section 22 is empty), the connection management section 21 waits until it becomes possible to read data from the data temporary storage section 22, and notifies to the second processing section 14 a result of reading data from the temporary storage section 22. In the stream data processing apparatus 10 comprising the connection management section 21 which

operates in this manner, the processing by the first processing section 13 may be halted until completion of data transmission, or the processing by second processing section 14 may be halted until completion of data reception, depending on the state of the data temporary storage section 22. In such situations, the stream data processing apparatus 10 is said to be operating in a "blocking mode".

[0022] In the stream data processing apparatus 10, when accessing the connection management section 21, either the data transmission section 24 or the data reception section 25 is used. The data transmission section 24 provides a data transmission function, which is realized via the connection management section 21. The data reception section 25 provides a data reception function, which is realized via the connection management section 21. The data transmission function provided by the data transmission section 24 and the data reception function provided by the data reception section 25 are implemented by means of a unified interface. In the stream data processing apparatus 10, the first processing section 13 utilizes the data transmission section 24 when transmitting data to the second processing section 14 via the connection management section 21. The second processing section 14 utilizes the data reception section 25 when receiving data from the first processing section 13 via the connection management section 21.

[0023] Specific examples of the data transmission section 24

and the data reception section 25 will be described. For example, a case will be described where the stream data processing apparatus 10 is implemented as a multi-thread system which operates on a certain operating system (hereinafter referred to as "OS"). In this case, the stream data processing apparatus 10 is designed by utilizing an inter-thread communication function which is supported by the OS. Note that, by providing the designer of the stream data processing apparatus 10 with a system library for utilizing the functions of the OS, the designer of the stream data processing apparatus 10 can design threads which perform inter-thread communications by simply using the provided system library, without knowing the details of the operational principles of the OS. Referring to FIG. 1, the connection management section 21 would correspond to a portion of the OS that supports the inter-thread communication function, and the data transmission section 24 and the data reception section 25 would correspond to a portion of the system library that is related to the use of the inter-thread communication function. Thus, by utilizing the data transmission section 24 and the data reception section 25, the designer of the stream data processing apparatus 10 will be able to design the first processing section 13 and the second processing section 14 so as to be capable of performing data exchanges via the connection management section 21, without knowing the operational details of the connection management section 21.

[0024] In a normal operation state, the constituent elements

of the stream data processing apparatus 10 operate as described above. Thus, the stream data processing apparatus 10 performs a two-step pipeline processing for the inputted stream data. For example, under the assumptions that the data input section 12 is a storage device which stores an MPEG video stream~~;~~, the first processing section 13 is an MPEG video decoder~~;~~, the second processing section 14 is a renderer filter~~;~~, and the data output section 15 is a display device~~.~~. According to this configuration the stream data processing apparatus 10 performs the process~~ing~~ of decoding and displaying an MPEG video stream in two steps.

[0025] In the stream data processing apparatus 10, it may sometimes become necessary to change the subject of processing, for example, when a command is inputted from the user, or when an abnormality occurs within the apparatus. As used herein, a "change of the subject of processing" refers to a seek (i.e., commencing processing at a different position in a given stream of data which is being processed), or a change of stream data (i.e., commencing processing for a stream of data which is different from a given stream of data which is being processed), for example.

[0026] When it becomes necessary to change the subject of processing, the control section 11 outputs a control signal 31 for instructing a change of the subject of processing to the first processing section 13, and either simultaneously or slightly earlier or later than this, outputs a control signal 32 instructing the same to the second processing section 14. If the control signal

31 is received from the control section 11, the first processing section 13 outputs a transmitting-end clear request 41 to the connection management section 21 by using the data transmission section 24, in order to erase the data stored in the data temporary storage section 22. Similarly, if the control signal 32 is received from the control section 11, the second processing section 14 outputs a receiving-end clear request 42 to the connection management section 21 by using the data reception section 25.

[0027] Based on the transmitting-end clear request 41 or the receiving-end clear request 42, the connection management section 21 operates as follows. The connection management section 21 is in a normal operation state until receiving the transmitting-end clear request 41 or the receiving-end clear request 42. In a normal operation state, data which is transmitted from the first processing section 13 is written to the data temporary storage section 22, and data to be received by the second processing section 14 is read from the data temporary storage section 22. If a transmitting-end clear request 41 is received in a normal operation state, the connection management section 21 transitions to a receiving-end clear wait state, in which the data to be received by the second processing section 14 is read from the empty data storage section 23. If a receiving-end clear request 42 is received in a normal operation state, the connection management section 21 transitions to a transmitting-end clear wait state, in which the data which is transmitted from the first processing section

13 is written to the empty data storage section 23. If a receiving-end clear request 42 is received in a receiving-end clear wait state or if a transmitting-end clear request 41 is received in a transmitting-end clear wait state, the connection management section 21 transitions to a normal operation state. The details of the operation of the connection management section 21 will be described later (with reference to FIG. 4).

[0028] The empty data storage section 23 is a buffer which erases any data that is written thereto, and which returns empty data to any device which attempts to read data therefrom. In other words, any data that is written to the empty data storage section 23 will not be stored in the empty data storage section 23. Even if a data read from the empty data storage section 23 is attempted, only empty data (i.e., 0 bytes) can be read from the empty data storage section 23. A specific example of the empty data storage section 23 is a null device which is supported by some OS's.

[0029] Hereinafter, referring to FIGS. 2 to 4, the details of the operation of the stream data processing apparatus 10 will be described. FIG. 2 is a flowchart illustrating the operation of the stream data processing apparatus in the case where a commencement of processing is instructed in a processing stopped state. FIG. 3 is a flowchart illustrating the operation of the stream data processing apparatus in the case where a change of the subject of processing is instructed in a normal operation state. FIG. 4 is a flowchart illustrating the operation of the connection

management section 21.

[0030] If a commencement of processing is instructed in a processing stopped state, the stream data processing apparatus 10 performs the process shown in FIG. 2. The process shown in FIG. 2 is started when the user inputs a begin-process command to the stream data processing apparatus 10. If the user inputs a begin-process command, the control section 11 receives the inputted begin-process command (step S101). Next, the control section 11 outputs a control signal 31 which instructs a commencement of processing to the first processing section 13, and either simultaneously or slightly earlier or later than this, outputs a control signal 32 instructing the same to the second processing section 14 (step S102).

[0031] After step S102 is executed, the first processing section 13 executes steps S111 to S114, and the second processing section 14 executes steps S131 to S134. Also, the connection management section 21 performs a process shown in FIG. 4 (step S121) as described later, independently from the processes performed by the first processing section 13 and the second processing section 14. In other words, after step S102 is executed, the process by the first processing section 13, the process by the second processing section 14, and the process by the connection management section 21 are performed in parallel or concurrently in the stream data processing apparatus 10.

[0032] The first processing section 13 repeatedly executes

steps S111 to S114 as described below until its processing for the stream data is completed. From the data input section 12, the first processing section 13 receives as an input the data contained in the stream data to be processed (step S111). Next, the first processing section 13 subjects the inputted data to a process which is specific to the first processing section 13 (step S112). Then, by using the data transmission section 24, the first processing section 13 transmits the processed data to the connection management section 21 (step S113). Then, the first processing section 13 determines whether its processing for the stream data has been completed or not (step S114). If it is determined that the processing is complete, the first processing section 13 stops processing. On the other hand, if it is determined that processing is incomplete, the first processing section 13 proceeds to step S111.

[0033] The second processing section 14 repeatedly executes steps S131 to S134 as described below until its processing for the stream data is completed. By using the data reception section 25, the second processing section 14 receives from the connection management section 21 the data which has been transmitted from the first processing section 13 (step S131). Next, the second processing section 14 subjects the received data to a process which is specific to the second processing section 14 (step S132). Then, the second processing section 14 outputs the processed data to the data output section 15 (step S133). Then, the second processing

section 14 determines whether its processing for the stream data has been completed or not (step S134). If it is determined that the processing is complete, the second processing section 14 stops processing. On the other hand, if it is determined that processing is incomplete, the second processing section 14 proceeds to step S131.

[0034] If a change of the subject of processing is instructed in a normal operation state, the stream data processing apparatus 10 performs the process shown in FIG. 3. The process shown in FIG. 3 is started when the user inputs a seek command, a stream data change command, or the like. If the a seek command, a stream data change command, or the like is inputted by the user, the control section 11 receives the inputted command (step S141). Next, the control section 11 outputs a control signal 31 instructing a change of the subject of processing to the first processing section 13, and either simultaneously or slightly earlier or later than this, outputs a control signal 32 instructing the same to the second processing section 14 (step S142).

[0035] After step S142 is executed, the first processing section 13 outputs a transmitting-end clear request 41 to the connection management section 21 by using the data transmission section 24 (step S110), and the second processing section 14 outputs a receiving-end clear request 42 to the connection management section 21 by using the data reception section 25 (step S130). At this time, it depends on the structure or operational state of the stream

data processing apparatus 10 which one of the transmitting-end clear request 41 and the receiving-end clear request 42 is outputted first.

[0036] After executing step S110, the first processing section 13 repeatedly executes steps S111 to S114 as in the case illustrated in FIG. 2. Also, after executing step S130, the second processing section 14 repeatedly executes steps S131 to S134, as in the case illustrated in FIG. 2. Henceforward, as in the case illustrated in FIG. 2, in the stream data processing apparatus 10, the process by the first processing section 13, the process by the second processing section 14, and the process by the connection management section 21 are performed in parallel or concurrently in the stream data processing apparatus 10.

[0037] As shown in FIG. 4, the connection management section 21 executes steps S201 and S202 in a normal operation state, executes steps S211 to S213 in a receiving-end clear wait state (i.e., a state which exists after a transmitting-end clear request 41 is received and until a receiving-end clear request 42 is received), and executes steps S221 to S223 in a transmitting-end clear wait state (i.e., a state which exists after a receiving-end clear request 42 is received and until a transmitting-end clear request 41 is received). In the following description, the data which is transmitted from the first processing section 13 will be referred to as "transmission data" , and the data to be received by the second processing section 14 will be referred to as

"data-to-be-received".

[0038] In a normal operation state, the connection management section 21 selects the data temporary storage section 22 both as a destination at which to write transmission data (hereinafter referred to as a "write destination for transmission data") and as a source from which to read the data-to-be-received (hereinafter referred to as a "read source of data-to-be-received"). Thus, in a normal operation state, the connection management section 21 writes transmission data to the data temporary storage section 22, and reads data-to-be-received from the data temporary storage section 22 (step S201). Next, the connection management section 21 determines whether a clear request has been received from the first processing section 13 or the second processing section 14 (step S202). If a transmitting-end clear request 41 has been received, the connection management section 21 proceeds to step S211; if a receiving-end clear request 42 has been received, the connection management section 21 proceeds to step S221; if neither clear request has been received, the connection management section 21 proceeds to step S201.

[0039] FIG. 5 is a diagram illustrating a manner in which data is transmitted or received at step S201. In FIG. 5, the write destination for transmission data and the read source of data-to-be-received are both set to be the data temporary storage section 22. The data temporary storage section 22 is illustrated as storing three pieces of data which have been transmitted from

the first processing section 13 but not yet received by the second processing section 14.

[0040] Referring back to FIG. 4, if step S202 finds that a transmitting-end clear request 41 has been received, the connection management section 21 erases the data which is stored in the data temporary storage section 22 (step S211). Next, the connection management section 21 selects the empty data storage section 23 as the read source of data-to-be-received, while still selecting the data temporary storage section 22 as the write destination for transmission data. Thus, in a receiving-end clear wait state, the connection management section 21 writes transmission data to the data temporary storage section 22, and reads data-to-be-received from the empty data storage section 23 (step S212). Next, the connection management section 21 determines whether a receiving-end clear request 42 has been received from the second processing section 14 (step S213). If a receiving-end clear request 42 has been received, the connection management section 21 proceeds to step S201; otherwise, control proceeds to step S212.

[0041] FIG. 6 is a diagram illustrating a manner in which data is transmitted or received at step S212. In FIG. 6, the write destination for transmission data is set to be the data temporary storage section 22, whereas the read source of data-to-be-received is set to be the empty data storage section 23. Any data which is stored in the data temporary storage section 22 is once erased

at step S211. In FIG. 6, the data temporary storage section 22 is illustrated as storing three pieces of data which were transmitted from the first processing section 13 after the data erasure but not yet received by the second processing section 14.

[0042] While the connection management section 21 is in a receiving-end clear wait state, the second processing section 14 will request the connection management section 21 to receive the rest of the data which the second processing section 14 was processing, despite the fact that the first processing section 13 has already changed its subject of processing. However, the "rest" of the data, which was stored in the data temporary storage section 22 while the connection management section 21 is in a normal operation state, has already been erased because the connection management section 21 has executed step S211. Moreover, while the connection management section 21 is in a receiving-end clear wait state, the first processing section 13 requests to the connection management section 21 that new data which emerged after the change of the subject of processing be transmitted, and so the connection management section 21 writes such transmission data to the data temporary storage section 22. Therefore, without providing a solution to this situation, the second processing section 14 would mistakenly receive the new data which emerged after the change of the subject of processing transmitted from the first processing section 13, and process such data as the rest of the data which the second processing section 14 was processing.

As a solution to this problem, as described above, the connection management section 21 sets the read source of data-to-be-received to be the empty data storage section 23 in a receiving-end clear wait state. As a result, while the connection management section 21 is in a receiving-end clear wait state, the second processing section 14 is prevented from receiving anything but empty data. Thus, the aforementioned problem is solved so that the stream data processing apparatus 10 is prevented from malfunctioning.

[0043] Referring back to FIG. 4, if step S202 finds that a receiving-end clear request 42 has been received, the connection management section 21 erases the data stored in the data temporary storage section 22 (step S221). Next, the connection management section 21 selects the empty data storage section 23 as the write destination for transmission data, while still selecting the data temporary storage section 22 as the read source of data-to-be-received. Thus, in a transmitting-end clear wait state, the connection management section 21 writes transmission data to the empty data storage section 23, and reads data-to-be-received from the data temporary storage section 22 (step S222). Next, the connection management section 21 determines whether a transmitting-end clear request 41 has been received from the first processing section 13 (step S223). If a transmitting-end clear request 41 has been received, the connection management section 21 proceeds to step S201; otherwise, control proceeds to step S222.

[0044] FIG. 7 is a diagram illustrating a manner in which data is transmitted or received at step S222. In FIG. 7, the write destination for transmission data is set to be the empty data storage section 23, and the read source of data-to-be-received is set to be the data temporary storage section 22. Any data which is stored in the data temporary storage section 22 is erased at step S221, and any data which is transmitted from the first processing section 13 after the data erasure is written to the empty data storage section 23. As a result, in FIG. 7, the data temporary storage section 22 is illustrated as storing no data.

[0045] While the connection management section 21 is in a transmitting-end clear wait state, the first processing section 13 as the transmitting-end processing section will request that the rest of the data which the first processing section 13 was processing be transmitted, despite the fact that the second processing section 14 has already changed its subject of processing. However, the second processing section 14 requests the connection management section 21 to receive new data which emerged after the change of the subject of processing. Therefore, without providing a solution to this situation, the second processing section 14 would mistakenly receive the rest of the data which the first processing section 13 was processing and transmitted, and process such data as new data which emerged after the change of the subject of processing. As a solution to this problem, as described above, in a transmitting-end clear wait state, the connection management

section 21 sets the write destination for transmission data to be the empty data storage section 23. As a result, while the connection management section 21 is in a transmitting-end clear wait state, the second processing section 14 is prevented from receiving anything but empty data. Thus, the aforementioned problem is solved so that the stream data processing apparatus 10 is prevented from malfunctioning.

[0046] FIG. 8 is a table illustrating the above-described operation (FIG. 4) of the connection management section 21. In the table of FIG. 8, the processes described in the upper left column and the lower right column are the processes to be performed in a normal operation state (step S201)~~,~~ the processes described in the lower left column are the processes to be performed in a receiving-end clear wait state (step S212)~~,~~ and the processes described in the upper right column are the processes to be performed in a transmitting-end clear wait state (step S222).

[0047] As described above, in accordance with the stream data processing apparatus of the present embodiment, in a normal operation state, the connection management section utilizes the data temporary storage section as a buffer for allowing the data transmitted from the first processing section (as the transmitting-end processing section) to be received by the second processing section (as the receiving-end processing section). If it becomes necessary to change the subject of processing due to a seek or change of stream data, etc., the first processing section

and the second processing section output a transmitting-end clear request and a receiving-end clear request, respectively, in response to an instruction from the control section. When either type of clear request is received in a normal operation state, the connection management section erases any data stored in the data temporary storage section, and sets the read source of data-to-be-received to be the empty data storage section in a receiving-end clear wait state, and sets the write destination for transmission data to be the empty data storage section in a transmitting-end clear wait state.

[0048] Thus, when the subject of processing is changed, the second processing section is prevented from mistakenly receiving any new data which emerged after the change of the subject of processing and processing it as the rest of the data which the second processing section 14 was processing, or receiving the rest of the data which the first processing section was processing and processing such data as new data which emerged after the change of the subject of processing. Thus, by properly exchanging data between a plurality of processing sections while achieving synchronization therebetween, the stream data processing apparatus can operate with a good overall integrity.

[0049] Moreover, each processing section only needs to respond to an instruction of a change of the subject of processing from the control section by outputting a clear request; other than that, the processing section can continue to properly exchange data by

requesting data transmission/reception. Thus, each processing section is capable of properly performing the exchange of data without having to pay attention to the data content or the state of another processing section with which data is exchange. As a result, the structure of each processing section can be simplified. In particular, in the case where each processing section is implemented in software, the software structure of each processing section can be simplified, thereby facilitating the design of the software components implementing the processing sections. A simpler software structure provides the advantage of reducing the amount of program code, which in turn makes for an improved software quality.

[0050] Moreover, each processing section exchanges data by using a data transmission provided by the data transmission section and a data reception function provided by the data reception section, which are implemented by means of a unified interface. As a result, the structure of each processing section is further simplified. In the case where each processing section is implemented in software, the software structure of each processing section can be even more simplified, and the reusability of each processing section is enhanced. Thus, in the case where each processing section is implemented in software, it is made easier to port the software component constituting each processing section into another system.

(second embodiment)

[0051] The stream data processing apparatus according to the first embodiment of the present invention is illustrated as erasing any data stored in the data temporary storage section when a transmitting-end clear request or a receiving-end clear request is outputted in a normal operation state. On the other hand, a stream data processing apparatus according to the second embodiment of the present invention is characterized in that, if a transmitting-end clear request is outputted first in a normal operation state, any data stored in the data temporary storage section is not erased, and in a receiving-end clear wait state, both new data and old data are stored in the data temporary storage section, in such a manner that the new data is distinguishable from the old data.

[0052] The stream data processing apparatus of the present embodiment has the same structure as that of the stream data processing apparatus of the first embodiment (as illustrated in FIG. 1), and performs the same processes as those performed by the stream data processing apparatus of the first embodiment in the case where a commencement of processing is instructed in a processing stopped state (FIG. 2) or a change of the subject of processing is instructed in a normal operation state (FIG. 3). Therefore, any description of these aspects will be omitted in the present embodiment. The stream data processing apparatus of the present embodiment differs from the stream data processing apparatus of the first embodiment with respect to the operation

of the connection management section 21 in a receiving-end clear wait state.

[0053] FIG. 9 is a flowchart illustrating the operation of the connection management section 21 of the stream data processing apparatus according to the second embodiment of the present invention. As shown in FIG. 9, the connection management section 21 executes steps S201 and S202 in a normal operation state, executes steps S231 to S233 in a receiving-end clear wait state, and executes steps S221 to S223 in a transmitting-end clear wait state. Since the process performed in a normal operation state and the process performed in a transmitting-end clear wait state are the same as those performed by the stream data processing apparatus of the first embodiment, the descriptions thereof are omitted (see FIG. 4). In the following description, any data which is stored in the data temporary storage section 22 when a transmitting-end clear request 41 has been outputted will be referred to as "old data", whereas any data which is transmitted from the first processing section 13 while the connection management section 21 is in a receiving-end clear wait state will be referred to as "new data". A region in the data temporary storage section 22 in which old data is stored will be referred to as an "old data region", whereas a region in the data temporary storage section 22 in which no old data is stored will be referred to as a "new data region".

[0054] As shown in FIG. 9, when step S202 finds that a

transmitting-end clear request 41 has been received, the connection management section 21 proceeds to step S231 without erasing the old data. In this case, the connection management section 21 selects the new data region as the write destination for transmission data, and selects the old data region as the read source of data-to-be-received. Note, however, that the old data region is selected as the read source of data-to-be-received only if there is any unread old data. Once old data is no longer present, the empty data storage section 23 is selected as the read source of data-to-be-received. Thus, in a receiving-end clear wait state, the connection management section 21 writes transmission data in the new data region, and reads data-to-be-received from the old data region (or from the empty data storage section 23 if no old data is present) (step S231).

[0055] Next, the connection management section 21 determines whether a receiving-end clear request 42 has been received from the second processing section 14 (step S232). If a receiving-end clear request 42 has been received, the connection management section 21 proceeds to step S233; otherwise, control proceeds to step S231. At step S233, the connection management section 21 erases any old data present (step S233), and thereafter proceeds to step S201.

[0056] FIGS. 10 and 11 are diagrams illustrating manners in which data is transmitted or received at step S231. FIG. 10 illustrates the case where old data is present. In FIG. 10, the

data temporary storage section 22 is illustrated as storing two pieces of old data (OLD_DATA5 and OLD_DATA6) and one piece of new data (NEW_DATA1), these two kinds of data being stored so as to be distinguishable from each other. If the first processing section 13 requests transmission of a next piece of new data in the state as shown in FIG. 10, the connection management section 21 writes such data to the new data region in the data temporary storage section 22 (e.g., in the next slot to the new data NEW_DATA1). If the second processing section 14 requests receiving a next piece of old data (OLD_DATA5) in the state as shown in FIG. 10, the connection management section 21 reads such data from the old data region in the data temporary storage section 22, and transmits it to the second processing section 14.

[0057] FIG. 11 illustrates the case where old data is no longer present. If the second processing section 14 requests receiving two pieces of old data and the first processing section 13 requests transmission of one new piece of data in the state shown in FIG. 10, a state shown as in FIG. 11 results. In FIG. 11, the data temporary storage section 22 is illustrated as storing two pieces of new data (NEW_DATA1 and NEW_DATA2). If the first processing section 13 requests transmission of a next piece of new data in the state shown in FIG. 11, the connection management section 21 writes such data to the new data region in the data temporary storage section 22 (e.g., in the next slot to the new data NEW_DATA2). If the second processing section 14 requests receiving a next piece of

old data in the state shown in FIG. 11, the connection management section 21 reads empty data from the empty data storage section 23. Since the stream data processing apparatus 10 operates in a blocking mode, the processing by the second processing section 14 will be halted until the completion of data reception.

[0058] As in the first embodiment, while the connection management section 21 is in a receiving-end clear wait state, the second processing section 14 will request receiving old data, despite the fact that the first processing section 13 has already changed its subject of processing. Remember that in the first embodiment, the connection management section 21 executes step S211, so that old data would already have been erased. However in the present embodiment, the old data has not been erased at this point. Moreover, in a receiving-end clear wait state, the connection management section 21 writes transmission data as new data to the data temporary storage section 22, and reads as data-to-be-received the old data stored in the data temporary storage section 22. As a result, while the connection management section 21 is in a receiving-end clear wait state, the second processing section 14 can receive the old data, if any, but is prevented from receiving anything but empty data once old data is no longer present. Thus, the second processing section 14 can properly receive the old data stored in the data temporary storage section 22 and process it as the rest of the data which the second processing section 14 was processing.

[0059] FIG. 12 is a table illustrating the above-described operation (FIG. 9) of the connection management section 21. In the table of FIG. 12, the processes described in the upper left column and the lower right column are the processes to be performed in a normal operation state (step S201)+, the processes described in the lower left column are the processes to be performed in a receiving-end clear wait state (step S231)+, and the processes described in the upper right column are the processes to be performed in a transmitting-end clear wait state (step S222).

[0060] As described above, in accordance with the stream data processing apparatus of the present embodiment, in a normal operation state or in a transmitting-end clear wait state, the connection management section performs the same operation as that performed by the stream data processing apparatus of the first embodiment. On the other hand, in a receiving-end clear wait state, the connection management section regards any data stored in the data temporary storage section when a transmitting-end clear request has been received, and sets the write destination for transmission data to be the new data region in the data temporary storage section, and sets the read source of data-to-be-received to be the old data region in the data temporary storage section (or the empty data storage section once old data is no longer present).

[0061] Therefore, the stream data processing apparatus of the present embodiment provides, in addition to the advantages provided

by the stream data processing apparatus of the first embodiment, an advantage in that the second processing section can properly perform processing to the old data stored in the data temporary storage section after a transmitting-end clear request is outputted and until a receiving-end clear request is outputted.

[0062] Various variants of the stream data processing apparatus according to each embodiment of the present invention are possible, such as follows.

The stream data processing apparatus of each embodiment as described above is illustrated as comprising two processing sections which perform two-step processing for inputted stream data. Alternatively, the stream data processing apparatus may comprise three or more processing sections to perform three or more steps of processing for inputted stream data. In this case, the stream data processing apparatus may realize data exchanges via the connection management section between any and all processing sections.

[0063] FIG. 13 illustrates a variant stream data processing apparatus which performs three-step processing for inputted stream data. A stream data processing apparatus 50 shown in FIG. 13 comprises a first processing section 53, a second processing section 54, and a third processing section 55 to perform three steps of processing ~~for~~ the stream data which is inputted via the data input section 52, and outputs the processing result from the data output section 56. When a seek command or the like is inputted

from the user, the control section 51 outputs control signals 61, 62, and 63 to the first processing section 53, the second processing section 54, and the third processing section 55, respectively. The first processing section 53 employs a data transmission section 24a to transmit data to the second processing section 54. The second processing section 54 employs a data reception section 25a to receive data from the first processing section 53. Based on a transmitting-end clear request 41a and a receiving-end clear request 42a, the connection management section 21a performs data write and data read for the data temporary storage section 22a and the empty data storage section 23a, thereby allowing data which is transmitted from the first processing section 53 to be received by the second processing section 54. The data exchange between the second processing section 54 and the third processing section 55 is handled in a similar manner.

[0064] A stream data processing apparatus comprising three or more processing sections does not need to realize data exchanges via the connection management section between any and all processing sections. Instead, data exchanges via the connection management section may be realized only between some of the processing sections. For example, in the stream data processing apparatus 50 shown in FIG. 13, data exchanges via the connection management section may be performed between the first processing section 53 and the second processing section 54 while data exchanges which do not involve the connection management section may be

performed between the second processing section 54 and the third processing section 55.

[0065] Although the stream data processing apparatus of each embodiment above is illustrated as operating in a blocking mode, the stream data processing apparatus may operate in a non-blocking mode as described below. Further alternatively, the stream data processing apparatus may be structured so as to be able to select between a blocking mode operation and a non-blocking mode operation. In that case, the processing performed by each processing section can be arbitrarily controlled, depending on the operational state of the apparatus or the like, so that the processing performed by each processing section is halted or not halted in a data reception/transmission wait state.

[0066] In a stream data processing apparatus operating in a non-blocking mode, if transmission data cannot be written to the data temporary storage section, or if data-to-be-received cannot be read from the data temporary storage section, the connection management section immediately notifies an error to the processing section from which the transmission data originated or the processing section for which the data-to-be-received is intended. In this case, after requesting a data transmission or reception via the connection management section, each processing section waits until the data transmission or reception is complete, by performing data processes such as poll or select. Therefore, even if another event occurs during the waiting, the processing section

can immediately perform a process related to that event. For example, if the user instructs an interruption of processing to the control section during the waiting, the processing section can immediately stop processing in response to the processing interruption instruction which has been outputted from the control section. Thus, each processing section can perform a process related to another event even in a data transmission/reception wait state. As a result, when an instruction is inputted from the user, the stream data processing apparatus can promptly respond to the user's instruction, without having to newly activate a thread for responding to the instruction.

[0067] In the above embodiments, it is stated that the data input section 12 and/or the data output section 15 may be a storage device such as a hard disk or a memory. More specifically, the data input section 12 and/or the data output section 15 may be constructed so as to write data to or read data from a removable recording medium such as a memory card. In accordance with a stream data processing apparatus comprising such a data input section 12 and such a data output section 15, it becomes possible to store the processing result obtained with a first apparatus to the removable recording medium, remove the recording medium from the first apparatus, and mount the recording medium to a second apparatus. In this manner, the second apparatus is enabled to continue from the processing by the first apparatus to reproduce stream data, perform a seek, or the like.

[0068] The stream data processing apparatus according to the present invention enables data transmission or reception between processing sections in such a manner that one processing section does not need to pay attention to the state of the other processing section. Thus, the stream data processing apparatus according to the present invention can be utilized as a device for processing stream data, e.g., reproduction device of audio data or video data.

[0069] While the invention has been described in detail, the foregoing description is in all aspects illustrative and not restrictive. It is understood that numerous other modifications and variations can be devised without departing from the scope of the invention.